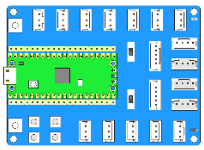
Lesson 7 How to use the line tracking module

In this lesson, we will learn how to use the line tracking module.

7.1 Components & Parts

Components	Quantity	Picture	Remark
Raspberry Pi Pico	1		
Pico Expansion board	1		
Line Tracking module	1		Not include in the Kit,you can prepared by yourself
5-Pin wires	1		

7.2 Working principle

"Line Tracking"--- On the white paper with black lines drawn, the black lines can be judged according to the intensity of the reflected light received due to the different reflection coefficients of the black lines and the white paper to light.

A more common detection method, infrared detection method, is used in the line tracking module. Infrared detection method is to use the characteristics of infrared rays that have different reflection properties on physical surfaces of different colors. During the running of the program, infrared light is continuously emitted to the ground. When the infrared light encounters the white ground, diffuse emission occurs, and the reflected light is received by the receiving tube; if it encounters a black line, the infrared light is absorbed, and the receiver of the line tracking module Signal cannot be received. The line tracking module we provide is a three-channel infrared tracking module, which contains 3 sets of sensors, each of which consists of an infrared emitting LED and an infrared receiver.

There are three Reflective Optical Sensors on Line Tracking Module. When the infrared light emitted by infrared diode shines on the surface of different objects, the sensor will receive light with different intensities after reflection.

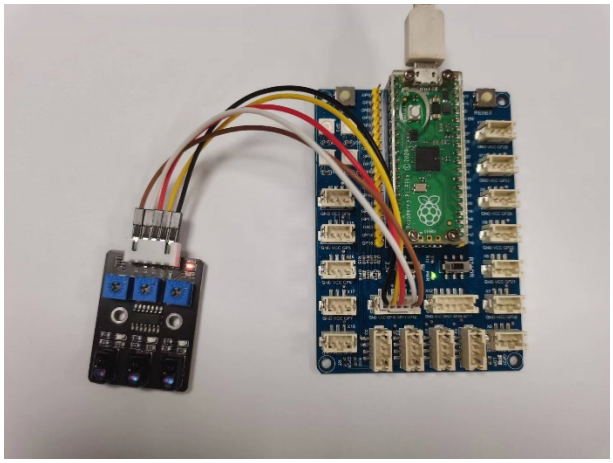
As we know, black objects absorb light better. So when black lines are drawn on the white plane, the sensor can detect the difference. The sensor can also be called Line Tracking Sensor.

Warning:

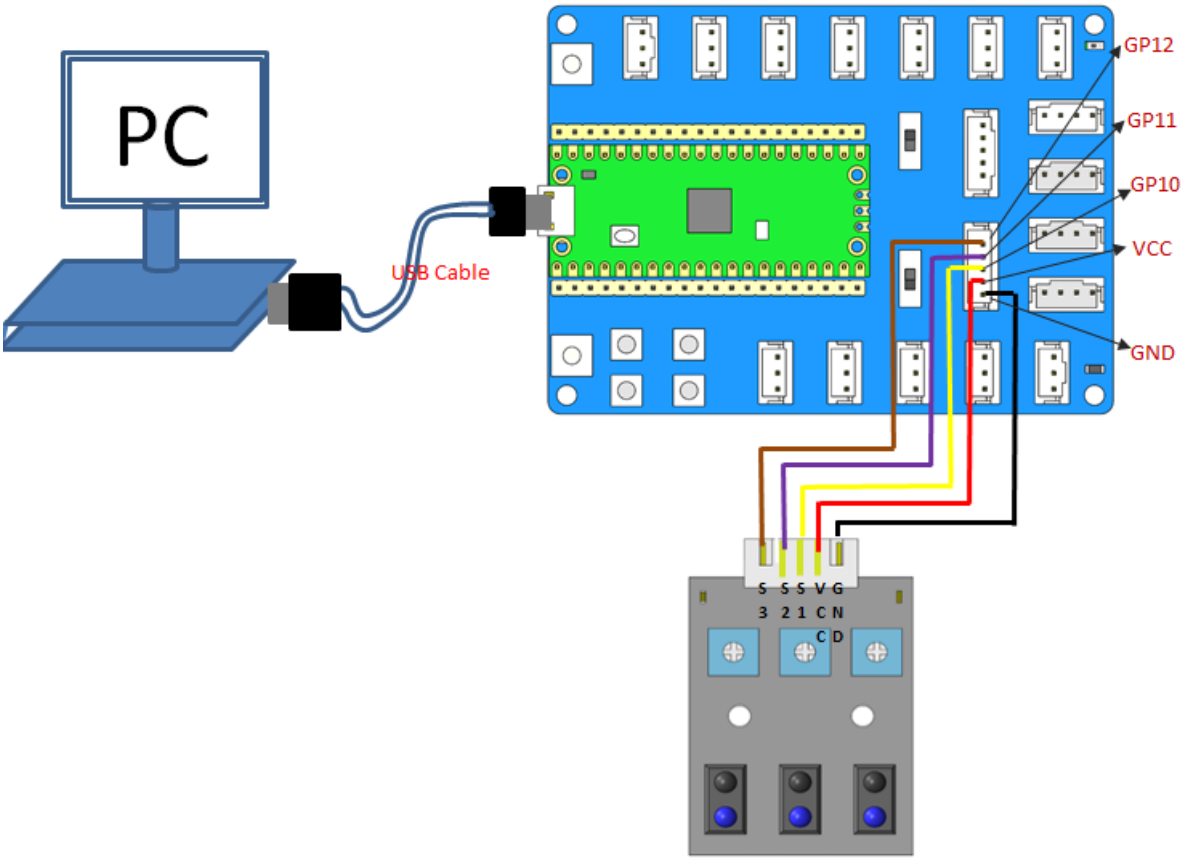
Reflective Optical Sensor (including Line Tracking Sensor) should be avoided using in environment with infrared interference, like sunlight. Sunlight contains a lot of invisible light such as infrared and ultraviolet. Under environment with intense sunlight, Reflective Optical Sensor cannot work normally.

7.3 Connection diagram

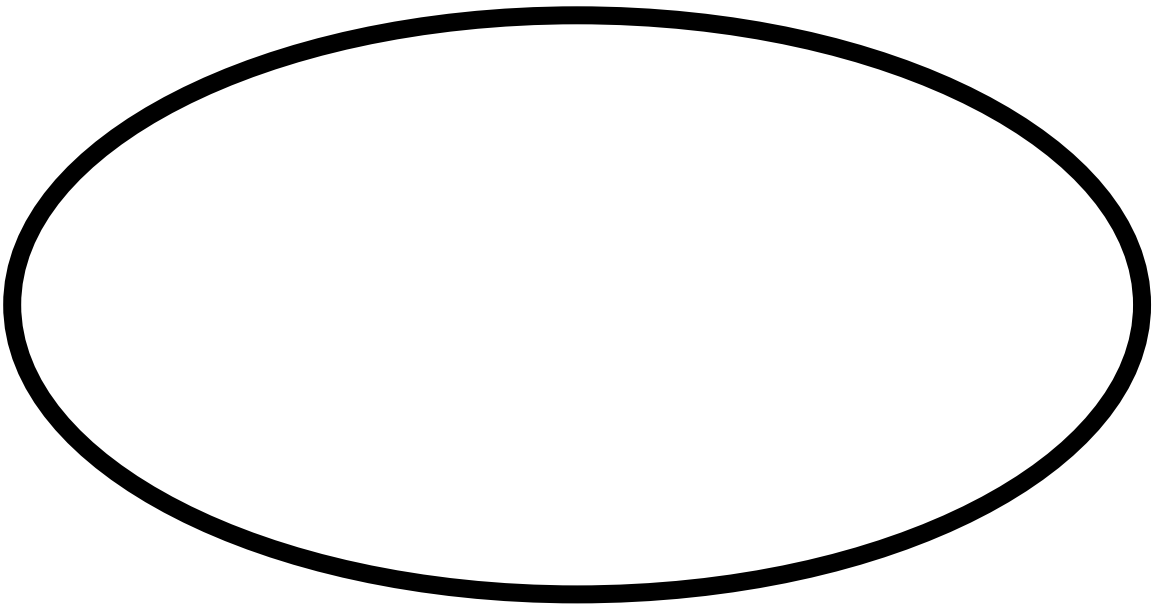
The line Tracking Module needs to be connected to the line Tracking Interface on the Pico Expansion board.



Hardware connection circuit



White paper with black lines: You can use black tape to stick on the white paper.

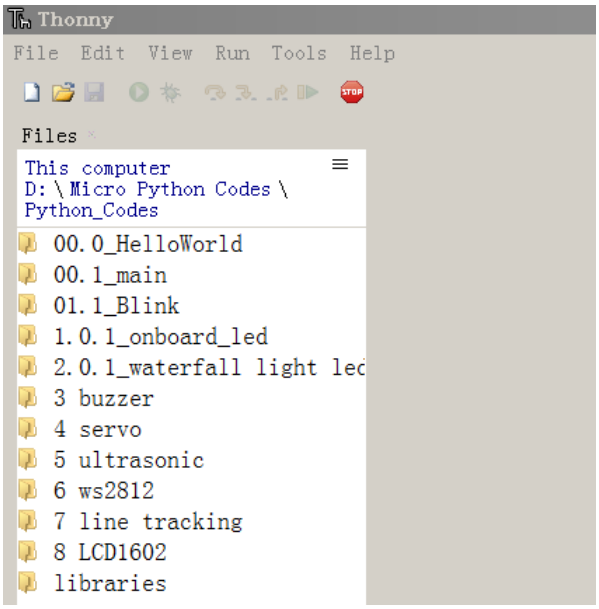


7.4 Run the program

7.4.1 Codes used in this tutorial are saved in "Pico Expansion Kit Tutorial\Lessons for Python\Python_Codes". You can move the codes to any location. For example, we save the codes in Disk(D) with the path of "D:/ Micro Python codes".

07_line_tracking

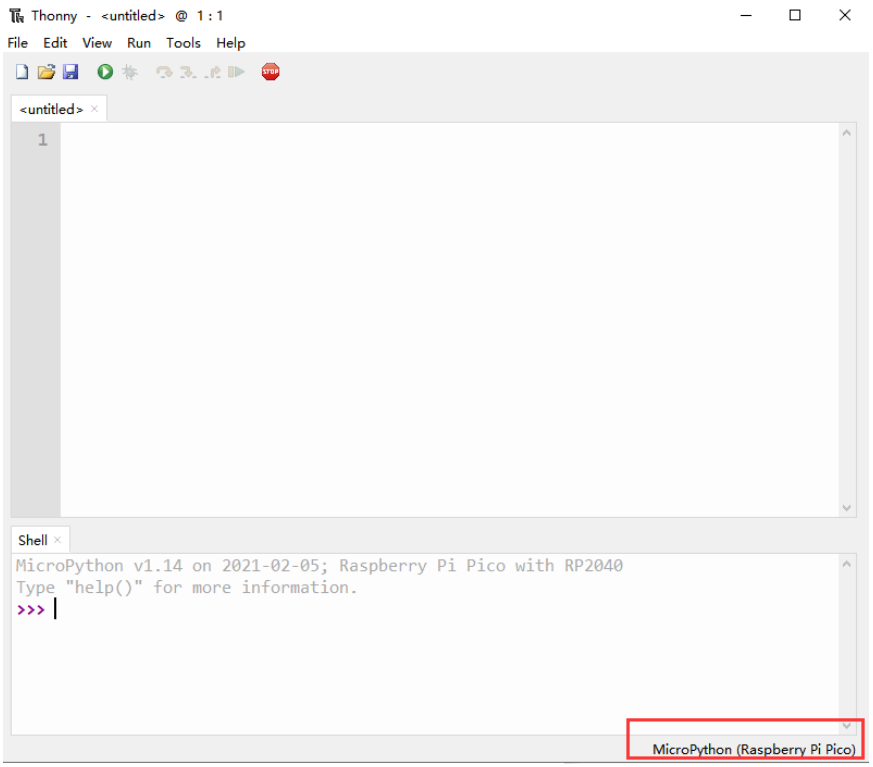
7.4.2 Open "Thonny", click "View"--"Files"--"This computer"--"D:"--"Micro Python Codes"--"Python_Codes".



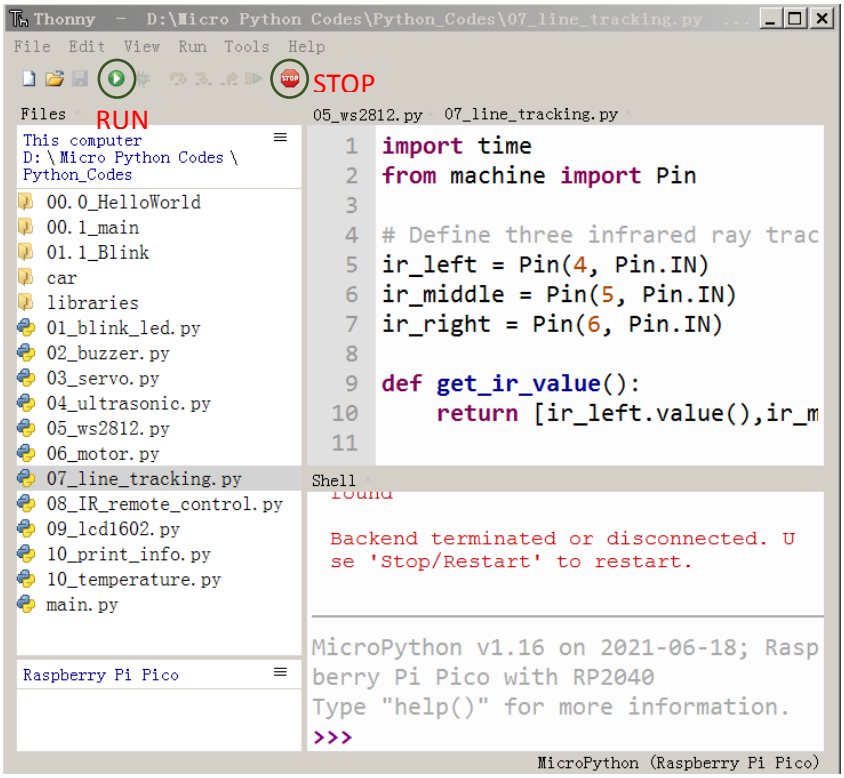
This computer: The file area of the personal computer.

Raspberry pi Pico: Pico file area, the code saved in Pico can be viewed in this area.

7.4.3 Confirm that "MicroPython (Raspberry Pi Pico)" is displayed in the lower right corner. If it is not, please click the font in the lower right corner to select "MicroPython(Raspberry Pi Pico)" mode.



7.4.4 Double-click the code "07_line_tracking.py" required for this course. The content of the code will be displayed in the interface on the right.

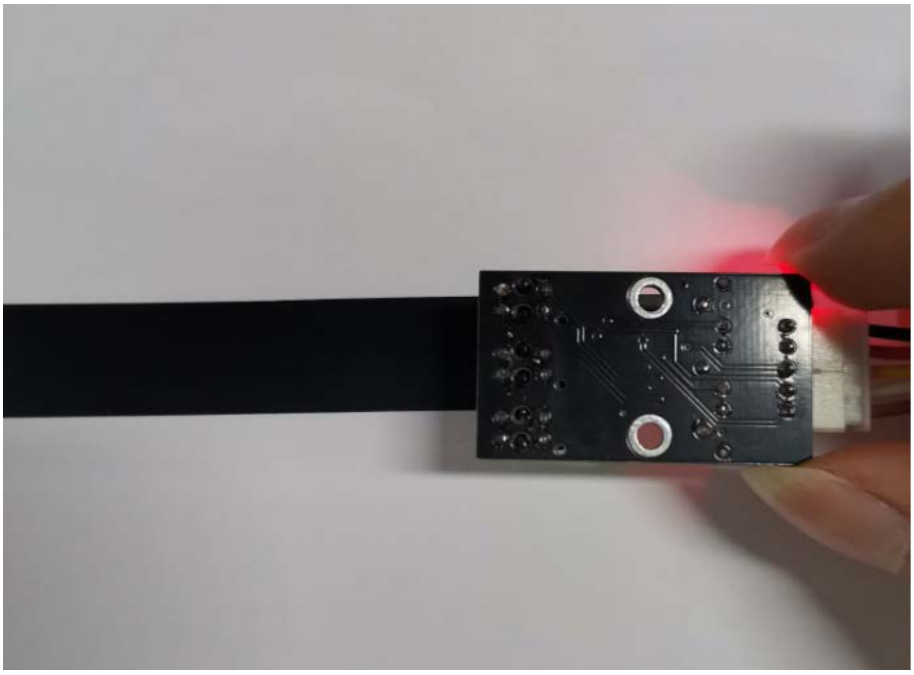


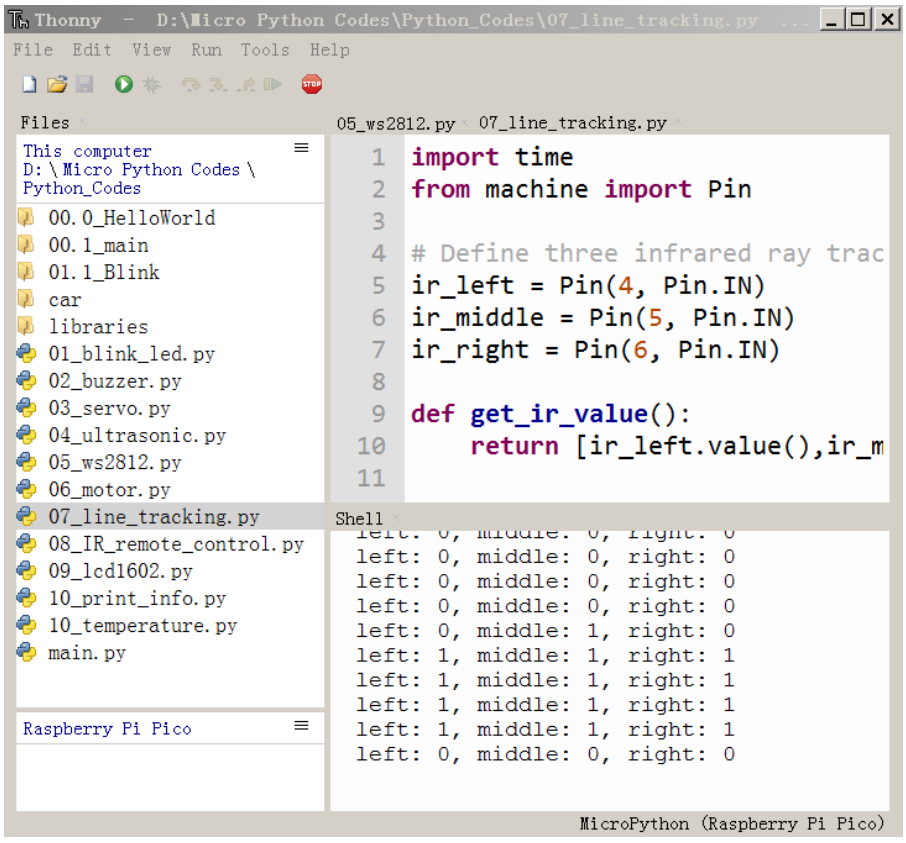
7.4.5 Click the Run button to run the program. You will see the detection results of 3 tracking channels on Thonny shell.

"1": The line tracking module detects a "black line", indicating that the infrared receiver does not receive data. The light from the infrared emitter is absorbed by the black lines.

"0": The line tracking module does not detect the "black line", which means that the infrared receiver has received the light from the infrared transmitter.

Note: If the test result is not satisfactory, you can try to fine-tune the blue potentiometer of the tracking module.





7.4.6 Click the stop button to stop the program,and the thonny shell will not print out information.

7.5 Code

07_line_tracking.py

```
1.  import time
2.  from machine import Pin
3.
4.  # Define three infrared ray tracking interface pins.
5.  ir_left = Pin(4, Pin.IN)
6.  ir_middle = Pin(5, Pin.IN)
7.  ir_right = Pin(6, Pin.IN)
8.
9.  def get_ir_value():
10.     return [ir_left.value(),ir_middle.value() ,ir_right.value()]
11.
12. def test():
13.     value = get_ir_value()
14.     print("left: {}, middle: {}, right: {}" .format(value[0], value[1], value[2]))
15.
16. if __name__ == '__main__':
17.     while True:
18.         test()
19.         time.sleep_ms(300)
```

7.6 What's Next?

THANK YOU for participating in this learning experience!

If you find errors, omissions or you have suggestions and/or questions about this Lesson, please feel free to contact us: cokoino@outlook.com

We will make every effort to make changes and correct errors as soon as feasibly possible and publish a revised version.

If you want to learn more about Arduino, Raspberry Pi, Smart Cars, Robotics and other interesting products in science and technology, please continue to visit our website. We will continue to launch fun, cost-effective, innovative and exciting products.

http://cokoino.com/

Thank you again for choosing Cokoino products.